

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

ANÁLISE DE JOGOS UTILIZANDO ALGORITMOS GENÉTICOS

ANÁLISE DE JOGOS UTILIZANDO ALGORITMOS GENÉTICOS

Monografia apresentada ao curso de Graduação em Bacharelado em Sistema de Informação da Faculdade Natálense para o Desenvolvimento do Rio Grande do Norte como requisito parcial para a obtenção do título de Bacharelado em Sistema de Informação.

Shirley Ingrid Dantas de Farias

Orientador:
Barbara Paraguaná Falcão Dantas Rocha

NATAL/RN
2004

**NATAL/RN
2004**

LIC. SHIRLEY INGRID DANTAS DE FARIAS
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

ANÁLISE DE JOGOS UTILIZANDO ALGORITMOS GENÉTICOS

Shirley Ingrid Dantas de Farias

Monografia apresentada e aprovada em 11 de dezembro de 2004 pela banca
examinadora composta pelos seguintes membros:

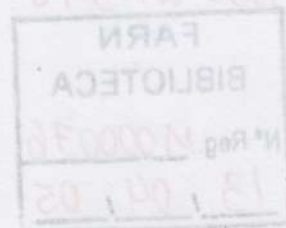
Monografia apresentada ao curso de
Graduação em bacharelado em Sistema
de Informação da Faculdade Natalense
para o Desenvolvimento do Rio Grande
do Norte como quesito parcial para a
obtenção do título em Bacharelado em
sistema de Informação.

Bartira Paraguaçu
Orientadora

Ysabel Roveda
Membro Examinador

Orientador:
Bartira Paraguaçu Falcão Dantas Rocha

NATAL/RN
2004



Catálogo na Publicação – Biblioteca da FARN
Setor de Processos Técnicos

FARIAS, Shirley Ingrid Dantas de

Análise de jogos utilizando algoritmos genéticos/

Shirley Ingrid Dantas de Farias. – Natal : FARN, 2004.

50p.

Orientadora: Bartira Paraguaçu Falcão Dantas Rocha

Monografia (Graduação em Sistemas de Informação) -

Faculdade Natalense Para o Desenvolvimento do Rio Grande do Norte

1. Algoritmos Genéticos – monografia 2. Jogos – monografia I. Título

RN/FARN/BC

CDU 004.421

RB- 21.345

FARN
BIBLIOTECA

Nº Reg 1000076

13 / 04 / 05

M
004.421

F238a

2004

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

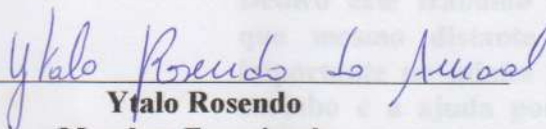
ANÁLISE DE JOGOS UTILIZANDO ALGORITMOS GENÉTICOS

Shirley Ingrid Dantas de Farias

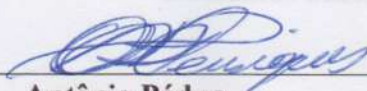
**Monografia apresentada e aprovada em 11 de dezembro de 2004 pela banca
examinadora composta pelos seguintes membros**



Bartira Paraguaçu Falcão Dantas Rocha
Orientador



Ytalo Rosendo
Membro Examinador



Antônio Pádua
Membro examinador

Natal, 11 de dezembro de 2004.

AGRADECIMENTOS

São muitas pessoas especiais a quem preciso agradecer, que tiveram grande importância na conclusão deste trabalho, na minha graduação e na minha vida.

Em primeiro lugar aos meus pais, Gêga e Fátima, por terem sido sempre maravilhosos e compreensivos, sabendo ao longo de minha vida, dar o carinho, amor e atenção necessária.

As minhas irmãs maravilhosas, Sheyla e Grace, que sempre me deram força e amor.

Ao meu namorado Allan, pela ajuda, paciência e compreensão.

Aos meus mestres, que se não fosse por eles, não estaria completando mais esta etapa. Mas, em especial, três deles tiveram muita importância na conclusão deste trabalho, os co-orientadores professor Antônio Padua e a professora Joazele Pinheiro, e a minha orientadora professora Berta Rocha.

É claro, os meus queridos colegas da FARN, aos quais considero amigos, não poderia esquecer tudo o que passamos nesses quatro anos de curso.

Dedico este trabalho a minha família que mesmo distante teve um papel importante na minha formação, com o carinho e a ajuda possível. E ao meu namorado Allan que teve paciência nos momentos difíceis.

AGRADECIMENTOS

São muitas pessoas especiais a quem preciso agradecer, que tiveram grande importância na conclusão deste trabalho, na minha graduação e na minha vida.

Em primeiro lugar aos meus pais, Graça e Farias, por terem sido sempre maravilhosos e compreensivos, sabendo ao longo de minha vida, dar o carinho, amor e atenção necessária.

As minhas irmãs maravilhosas, Sheyla e Grace, que sempre me deram força e amor.

Ao meu namorado Allan, pela ajuda, paciência e compreensão.

Aos meus mestres, que se não fosse por eles, não estaria completando mais esta etapa. Mas, em especial três deles tiveram suma importância na conclusão deste trabalho, os co-orientadores professor Antônio Pádua e a professora Joseane Pinheiro, e a minha orientadora professora Bartira Rocha.

E claro, os meus queridos colegas da FARN, aos quais considero amigos, não poderia esquecer tudo o que passamos nesses quatro anos de curso.

RESUMO

Este projeto relata o universo dos Algoritmos Genéticos como uma das áreas da Inteligência Artificial, é uma heurística. Após um estudo detalhado do mesmo, são apresentados dois estudos de caso mostrando a funcionalidade dos Algoritmos Genéticos em jogos.

Dois jogos são tidos como exemplo, o infantil, do *pingüim e da abelha*, e o *xadrez*. É feito ainda um estudo detalhado de como o algoritmo se comporta nos dois jogos, podendo assim, ser feito um comparativo entre um jogo simples, e um bem mais complexo, notando-se a importância de algoritmos Genéticos no estudo e implementação de problemas onde o universo de possibilidades tidas como resposta é muito amplo.

Palavras-chave:

Jogos, Algoritmo Genético, Inteligência Artificial.

ABSTRACT

This project tells the universe of the Genetic Algorithms as one of the areas of Artificial Intelligence, and a heuristic. After a study detailed of exactly, two studies of case are presented showing the functionality of the Genetic Algorithms in games. Two games are had as example, the infantile one, of the penguin and the bee, and the chess. A study detailed is made still of as the algorithm if it holds in the two games, thus being able, to be made a comparative degree between a simple game, and the well most complex one, noticing it importance of Genetic algorithms in the study and implementation of problems where the universe of had possibilities as reply is very ample.

1.1 JOGO DE COMPUTADOR	12
1.2 INTELIGÊNCIA ARTIFICIAL	14
1.2.1 Breve Histórico da Inteligência Artificial	14
1.2.2 Inteligência Artificial em Jogos	15
1.2.3 Campos de Aplicação da Inteligência Artificial	16
1.2.4 Modelos de Inteligência Artificial	17
1.3 HEURÍSTICA	19
1.4 ALGORITMOS GENÉTICOS	20
1.4.1 Principais Conceitos	21
1.4.2 Conceito de Algoritmo Genético	24
1.4.3 Operadores Genéticos	25
1.4.3.1 Seleção	25
1.4.3.2 Cruzamento Crossover	28
1.4.3.3 Mutação e Inversão	29
1.4.4 Critério de Parada	29
1.4.5 O Algoritmo	31
1.5 METODOLOGIA	32
1.6 ANÁLISE DE RESULTADOS	32
1.6.1 O JOGO DO PENGUIM E DA ABELHA	33
1.6.2 O JOGO DE XADREZ	38
1.7 CONCLUSÃO	47
1.8 REFERÊNCIAS BIBLIOGRÁFICAS	49

SUMÁRIO

1. INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.1.1 <i>Objetivo Geral:</i>	11
1.1.2 <i>Objetivos Específicos:</i>	11
1.2 JUSTIFICATIVA	11
2. REFERENCIAL TEÓRICO	12
2.1 JOGOS DE COMPUTADOR	12
2.2 INTELIGÊNCIA ARTIFICIAL	14
2.2.1 <i>Breve Histórico da Inteligência Artificial</i>	14
2.2.2 <i>Inteligência Artificial em Jogos</i>	15
2.2.3 <i>Campos de Aplicação da Inteligência Artificial</i>	16
2.2.4 <i>Modelos de Inteligência Artificial</i>	17
2.3 HEURÍSTICA	19
2.4 ALGORITMOS GENÉTICOS	20
2.4.1 <i>Principais Conceitos</i>	20
2.4.2 <i>Conceito de Algoritmo Genético</i>	21
2.4.3 <i>Operadores Genéticos</i>	24
2.4.3.1 <i>Seleção</i>	25
2.4.3.2 <i>Cruzamento Cross-Over</i>	25
2.4.3.3 <i>Mutação e Inversão</i>	28
2.4.4 <i>Critério de Parada</i>	29
2.4.5 <i>O Algoritmo</i>	29
3. METODOLOGIA	31
4. ANÁLISE DE RESULTADOS	32
4.1 O JOGO DO PINGÜIM E DA ABELHA	32
4.2 O JOGO DE XADREZ	38
5. CONCLUSÃO	47
6. REFERÊNCIAS BIBLIOGRÁFICAS	49

Lista de Figuras

Figura 1 - Exemplo de cromossomo.....	24
Figura 2 - Amostragem estocástica universal.....	25
Figura 3 - Exemplo de cruzamento uniponto.....	26
Figura 4 - Exemplo de cruzamento multiponto.....	27
Figura 5 - Exemplo de cruzamento uniforme.....	27
Figura 6 - Exemplo de mutação.....	29
Figura 7 - Exemplo de inversão.....	29
Figura 8 - Interface do jogo do pingüim e da abelha.....	32
Figura 9 - Cromossomo.....	33
Figura 10 - Fitness mais alto neste exemplo.....	35
Figura 11 - População inicial.....	36
Figura 12 - Números criados aleatórios.....	37
Figura 13 - Interface do jogo de xadrez.....	38
Figura 14 - Estrutura do cromossomo no xadrez.....	39
Figura 15 - Codificação dos movimentos possíveis do peão.....	40
Figura 16 - Codificação dos movimentos possíveis da torre.....	41
Figura 17 - Codificação dos movimentos possíveis do cavalo.....	41
Figura 18 - Codificação dos movimentos possíveis do bispo.....	42
Figura 19 - Codificação dos movimentos possíveis da rainha.....	42
Figura 20 - Codificação dos movimentos possíveis do rei.....	43
Figura 21 - Formação de uma população misturada.....	44

Introdução

Lista de tabelas

Tabela 1.....	34
Tabela 2.....	34
Tabela 3.....	34
Tabela 4.....	40

1. INTRODUÇÃO

A computação é uma área em constante evolução, devido a constante demanda por sistemas e serviços que satisfaçam as necessidades atuais, passando pelas grandes máquinas, os computadores de grande porte, até os dias atuais onde encontramos os micro computadores pessoais, que estão cada vez menores e com ótimo desempenho.

A *Inteligência Artificial* teve uma importante participação nesta evolução, com suas técnicas de construção de programas que permitam ao computador simular aspectos do comportamento da inteligência humana, tornando bem sucedidos o processamento e a execução de tarefas complexas, como os jogos.

Uma das diversas áreas abordadas pela computação, os jogos, necessita da utilização de métodos heurísticos, por possuir um campo de possibilidades muito amplo, e da Inteligência Artificial por precisar de auxílio na tomada de decisões, de acordo com as informações recebidas.

Os Algoritmos Genéticos representam uma importante área dentro da Inteligência Artificial que faz uso de heurísticas. Algoritmos genéticos definem técnicas que permitem a obtenção dos melhores resultados, ou o mais próximo destes, dentre um conjunto de resultados possíveis.

Isto é possível porque eles se baseiam na teoria da seleção natural de Darwin, permitindo ainda que o conjunto de soluções possa sofrer mutações e evoluções para um novo conjunto de soluções. Algoritmos genéticos são bastante úteis quando o espaço de soluções é grande. Que é o caso da maioria dos jogos, principalmente aqueles que são baseados em estratégias.

1.1 OBJETIVOS

1.1.1 Objetivo Geral:

O objetivo deste trabalho consiste em apresentar e discutir modelagens de jogos sob a ótica de algoritmos genéticos.

1.1.2 Objetivos Específicos:

- ◆ Apresentar em detalhes algoritmos genéticos, seus conceitos e técnicas;
- ◆ Apresentar o uso de Algoritmos Genéticos em situações reais de jogos;
- ◆ A partir da observação do funcionamento de algoritmos genéticos discutir sua adequabilidade no desenvolvimento de jogos.

1.2 JUSTIFICATIVA

A sociedade hoje está cada vez mais preocupada com o trabalho, buscando qualificação e atualização profissional, tomando o tempo sempre mais precioso. Com isso, principalmente nas grandes cidades, as pessoas estão adoecendo, e o stress é o grande vilão.

Os jogos foram criados exatamente com intuito de entretenimento, e sendo uma forma de lazer, torna-se importante para amenizar as tensões do cotidiano. Este é um dos motivos que leva tantos a se dedicarem a pesquisa, ao estudo e desenvolvimento de jogos em computador.

Existem também os jogos educativos computadorizados que possuem, como uma de suas principais vantagens, um grande potencial para o processo de ensino e aprendizagem, por despertarem naturalmente o interesse dos alunos em conceitos vistos somente em sala de aula.

Mas a importância de tanta dedicação aos jogos eletrônicos não limita-se apenas a divertir pessoas, ou ajudá-las a desenvolver o intelecto, pois as técnicas e estratégias utilizadas no seu desenvolvimento são utilizadas para outras funções como simulações de ambientes virtuais, que podem ser desenvolvidos para projetos delicados em grandes empresas.

2. REFERENCIAL TEÓRICO

2.1 JOGOS DE COMPUTADOR

Com o desenvolvimento da tecnologia da informática, os jogos atingiram um novo patamar, com a chegada dos jogos eletrônicos. A partir deste momento, o mundo do entretenimento começou a entrar numa nova era. Surgiram máquinas mais potentes e, conseqüentemente, jogos mais realistas, mais ricos em detalhes, cores, etc.

Um jogo de computador é um programa caracterizado por intensa interatividade e participação do usuário ou jogador, que utiliza sua habilidade com os dispositivos de entrada, seu raciocínio lógico e sua imaginação para alcançar um determinado objetivo. O objetivo do jogo pode ser uma tarefa simples e direta, como chegar em primeiro lugar em uma corrida ou eliminar do tabuleiro todas as peças do adversário. (SANTOS, 2002)

Na década de 50, a recém-fundada ciência da computação começou a desenvolver técnicas de inteligência artificial, com o objetivo de produzir programas de computador capazes de imitar funções complexas do intelecto humano. Dois cientistas norte-americanos tinham conseguido realizar a primeira demonstração de que isso era possível, desenvolvendo um programa capaz de provar teoremas de álgebra e geometria elementares. Em alguns casos, as provas desenvolvidas pela máquina foram inteiramente originais, ou seja, não coincidiam com as provas clássicas desenvolvidas pelos matemáticos. (SABBATINI, 1992)

Os jovens e irrequietos cientistas da época buscavam freneticamente outras áreas exclusivas do intelecto humano que pudessem ser utilizadas para comprovar os sucessos da Inteligência Artificial. E o que é mais complexo, tipicamente humano e intelectualmente desafiador, do que um bom jogo de xadrez? Assim, vários grupos partiram em busca desse objetivo. Em 1956, Alex Bernstein, um cientista da IBM, foi o primeiro a conseguir inventar um programa que jogasse xadrez em um tempo razoável, contornando o que John von Neumann denominou de explosão combinatória. Usando um computador IBM 704, Bernstein calculou que, se o programa fosse explorar todas as combinações possíveis de jogadas em cada nível (em média, cerca de 30 possibilidades por movimento), e considerando que um jogo tem cerca de 30 a 40 movimentos; existiria um número de jogos possíveis igual ao número 1 seguido de 120 zeros! Supondo que um computador fosse capaz de examinar um milhão de movimentos por segundo, ainda assim ele demoraria 10

elevado a 108 anos para jogar todos os jogos possíveis (um tempo maior que a atual idade do Universo). (SABBATINI, 1992)

O jogo é portanto, sob as suas duas formas essenciais de exercício sensório-motor e de simbolismo, uma assimilação da real à atividade própria, fornecendo a esta seu alimento necessário e transformando o real em função das necessidades múltiplas do eu..(PIAGET, 1976).

Os jogos de computador / entretenimento eletrônico representam atualmente um mercado mundial de bilhões de dólares. Este mercado tem sido pouco explorado tanto pela indústria como pelo mundo acadêmico no Brasil. A participação brasileira neste setor é limitada principalmente à importação e adaptação de produtos oriundos do exterior. As poucas iniciativas brasileiras de desenvolvimento baseiam-se na utilização de modelos já consagrados no exterior, sem apresentar grandes inovações ou números expressivos de vendas. (TATAI, 1992)

Segundo Peter Molyneux, um dos maiores projetistas de jogos da atualidade, "a inteligência artificial é o futuro dos jogos". Isto se torna mais evidente pois à medida em que as plataformas utilizadas aumentam em capacidade computacional (tanto no que concerne a parte lógica como gráfica), um maior volume de processamento se vê disponível para a inteligência dos jogos. Vê-se assim o foco tecnológico da indústria de jogos passando de um nível gráfico e se direcionando para as técnicas de inteligência artificial (sem perder-se de vista o foco principal, que é a jogabilidade). Nesse contexto, a utilização de recursos de inteligência artificial, não antes apresentados, constitui uma grande vantagem competitiva. É interessante notar que cada vez mais estão sendo utilizadas técnicas mais avançadas da inteligência artificial em jogos comerciais como: redes neurais, algoritmos genéticos, lógica fuzzy, sistemas especialistas, etc. Apesar da grande maioria dos jogos ainda ter sua inteligência baseada sobre métodos mais tradicionais, tais como máquinas de estado, sistemas de regras e utilização de scripts, há uma forte tendência no uso de técnicas inteligentes. (TATAI, 1992)

Do ponto de vista acadêmico, por outro lado, os jogos de computador apresentam-se como uma excelente plataforma para o teste e validação de novas metodologias e algoritmos, pois encontram nos ambientes de jogos a riqueza e a complexidade de ambientes sofisticados, proporcionando aplicações ideais para testar procedimentos complexos. Assim, podemos explorar desde jogos tradicionais, tais como xadrez, damas,

go-moku, entre outros, bem como os ambientes de realidade virtual proporcionados pelos jogos mais modernos às novas técnicas da Inteligência Artificial. Tais jogos apresentam novos desafios e oportunidades sobre o ponto de vista da inteligência artificial, devido ao alto nível de interação com o usuário e por possuírem ambientes ricos e dinâmicos, aproximando-os assim do mundo real, mas mantendo-os ainda sob ambiente controlado.

Os jogos, principalmente os de tabuleiro como damas, xadrez, go, etc, além de apresentarem todas as características de um domínio adequado para a experimentação da Inteligência Artificial, exercem ainda um grande atrativo devido ao seu aspecto lúdico. Programas para jogos foram muito importantes para a área da Inteligência Artificial, não tanto pelos resultados obtidos, mas pelo esclarecimento trazido a respeito da importância relativa dos diversos aspectos envolvidos em uma atividade inteligente. (BITTENCOURT, 1996)

2.2 INTELIGÊNCIA ARTIFICIAL

O que chamamos hoje de Inteligência Artificial (IA) é um ramo da ciência da computação ao mesmo tempo recente, oficialmente a IA nasceu em 1956, mas muito antigo, pois a IA foi construída a partir de idéias filosóficas, científicas e tecnológicas herdadas de outras ciências, algumas tão antigas quanto a lógica, com seus 23 séculos. (BITTENCOURT, 1996)

2.2.1 Breve Histórico da Inteligência Artificial

Os primeiros estudos sobre inteligência artificial surgiram na década de 40, que foi marcada pela II Guerra Mundial. Este fato resultou na necessidade de desenvolver uma tecnologia voltada para a análise de balística, quebra de códigos e cálculos para projetar a bomba atômica. Surgia, assim, os primeiros grandes projetos de construção de computadores, assim chamados por serem máquinas utilizadas para fazer cálculos (cômputos). (LIMA, 2004)

Após a II Guerra Mundial o computador não ficou restrito ao uso militar e científico, começou a ser gradualmente utilizado em empresas, indústrias, universidades.

etc. A diversidade de aplicações estimulou pesquisas de software, hardware e linguagens de programação.

O desenvolvimento do computador, primeiramente impulsionado pela aplicabilidade militar e posteriormente comercial, mostrou-se viável. Seu rápido progresso, desde o surgimento dos primeiros computadores eletrônicos (1943 - Collossus, na Inglaterra e 1946 - ENIAC, nos Estados Unidos) até o surgimento dos microcomputadores (na década de 70) demonstra que essa área recebeu grandes investimentos. (LIMA, 2004)

O segundo grande passo foi dado nos Estados Unidos, em 1956, quando John McCarthy reuniu em uma conferência proferida ao Dartmouth College, na Universidade de New Hampshire, vários pesquisadores de renome para estudar o que foi denominado por Minsky, McCarthy, Newell e Simon de Inteligência Artificial (IA), expressão utilizada para designar um tipo de inteligência construída pelo homem para dotar a máquina de comportamentos inteligentes. (LIMA, 2004)

A partir da estruturação desse novo campo do conhecimento, o fenômeno da inteligência começou a ser pesquisado de forma intensa. Vários esforços foram e têm sido feitos no sentido de simular os tipos de raciocínios utilizados pelo ser humano e implementá-los no computador por meio da IA.

2.2.2 Inteligência Artificial em Jogos

A Inteligência Artificial estuda os processos de raciocínio e ponderação do cérebro humano, e procura simulá-los através de algoritmos e heurísticas, com o objetivo de criar programas que imitem a maneira de pensar dos seres humanos. Projetando esse objetivo ao limite, esperamos que, no futuro, seja possível interagir com um programa de computador e não saber se nosso interlocutor é de fato a máquina ou uma outra pessoa. (SANTOS, 2002).

É exatamente isso que o jogador espera do universo de um jogo: que os personagens, o ambiente e as reações sejam o mais parecido possível com os do mundo real. E, além disso, devido ao impressionante desenvolvimento da computação gráfica para jogos, que os tornou muito mais visualmente realísticos, o jogador passou a ter expectativas de que o mesmo grau de realismo seja observado no comportamento dos personagens do jogo.

Para atingir esse realismo, torna-se cada vez mais comum nos jogos a implementação de algoritmos clássicos das mais variadas áreas da Inteligência Artificial, como Busca de Caminhos, Processamento de Linguagem Natural, Aprendizado de Máquina, Aquisição de Conhecimento, Comportamento em Grupo, Tomada de Decisões e Agentes Inteligentes. Dependendo do tipo de jogo e da complexidade dos personagens envolvidos, podem ser usados vários desses algoritmos: entidades que se locomovem livremente no espaço do jogo utilizam algoritmos de Busca de Caminhos; personagens cujo diálogo com o jogador é determinado dinamicamente utilizam técnicas de Processamento de Linguagem Natural; jogos em que o nível de dificuldade é modificado dinamicamente de acordo com a habilidade do jogador utilizam procedimentos de Aprendizado de Máquina; jogos em que os personagens agem em conjunto, coordenando suas ações para um objetivo comum, utilizam técnicas de Comportamento em Grupo. (SANTOS, 2002)

2.2.3 Campos de Aplicação da Inteligência Artificial

Existem vários campos de estudo dentro da IA com o propósito de dotar a máquina de capacidade de raciocínio, aprendizado e autoaperfeiçoamento. Alguns desses campos são: (LIMA, 2004)

- **Processamento de Linguagem Natural** - É o estudo voltado para a construção de programas capazes de compreender a linguagem natural (interpretação) e gerar textos. A Geração de linguagem Natural é a produção de textos por um programa a partir de um conteúdo semântico representado internamente no próprio programa. Objetiva aperfeiçoar a comunicação entre as pessoas e os computadores.
- **Reconhecimento de Padrões** - É uma das áreas de pesquisa bem avançadas da IA. A capacidade de reconhecimento de padrões permite ao programa reconhecer a fala em linguagem natural, os caracteres digitados e a escrita (ex.: assinatura). Os scanners, por exemplo, utilizam programas de reconhecimento óptico desenvolvidos pelas pesquisas em IA.
- **Visão de Computador** - Busca desenvolver formas do computador trabalhar com a visão bidimensional e tridimensional.

- Programação de Jogos - É o estudo voltado para a construção de programas de jogos envolvendo raciocínio. Os jogos computadorizados são um grande sucesso, principalmente quando as estratégias usadas desafiam as habilidades do jogador, prendendo-o ainda mais ao jogo. O jogo de xadrez, por exemplo, foi utilizado para as primeiras experiências em programação do raciocínio artificial, onde o computador se tornou capaz de analisar milhões de jogadas por segundo para tentar derrotar o adversário. Além de analisar as jogadas, os programas utilizam um método heurístico que consiste na utilização de uma árvore de busca, a mesma possui ramificações a partir de certos nós, que representam pontos de decisão no caminho a tomar, com um certo número de etapas, para chegar a um objetivo. Deste modo, ele pode analisar vários nós, de acordo com a situação atual do jogo, e escolher o melhor caminho (o mais curto ou menos arriscado).
- Robótica - É o campo de estudo voltado para desenvolver meios de construir máquinas que possam interagir com o meio (ver, ouvir e reagir aos estímulos sensoriais). A expressão robô vem do tcheco robota, significa trabalhador, foi criada por Karel Capek, em 1917. O primeiro robô industrial do mundo, batizado de UNIMATE, surgiu em 1962.
- Aprendizado - Existem programas de IA que conseguem aprender certos fatos por meio da experiência, desde que esse conhecimento possa ser representado de acordo com o formalismo adotado pelo programa.

2.2.4 Modelos de Inteligência Artificial

Os programas desenvolvidos através da IA caracterizam-se pelo fato de objetivarem uma interação com o usuário ou seu ambiente (sistema) por meio da simulação de atitudes e reações humanas que envolvem a compreensão, análise, planejamento, tomada de decisão, aprendizado, etc. Estes são exemplos de técnicas já exploradas: (LIMA, 2004)

- Algoritmos Genéticos (AG) - É um modelo para o aprendizado da máquina, baseado na Seleção Natural do inglês Charles Darwin (1809-1882), criador da teoria evolucionista, segundo a qual somente os mais aptos sobrevivem. Algoritmo

genético é um método utilizado pelos Algoritmos Evolutivos, que inclui o estudo dos algoritmos genéticos, estratégia de evolução, programação evolutiva e sistemas classificatórios. Os algoritmos genéticos foram criados por Jonh Holland (1975) e objetivam emular operadores genéticos, como crossing-over, mutação e reprodução da mesma forma como é observado na natureza. Isso é feito criando-se dentro da máquina uma população de indivíduos representados por cromossomas. Os indivíduos passam por um processo simulado de evolução, seleção e reprodução gerando uma nova população.

- Programação Evolutiva - Campo da IA concebido por Lawrence J. Fogel (1960), assemelha-se aos algoritmos genéticos, sendo que é dada maior ênfase na relação comportamental entre os parentes e seus descendentes. As soluções para os problemas são obtidas por meio de tentativas e transmitidas para a nova população (simulada em programas).
- Lógica FUZZY - Também denominada de Conjuntos Difusos ou Lógica Nebulosa. Foi estruturada por Lofti Zadeh da University of Califórnia, no ano de 1965. É uma metodologia que serve para representar, manipular e modelar informações incertas.
- Sistemas Baseados em Conhecimento - São sistemas que implementam comportamentos inteligentes de especialistas humanos.
- Programação Genética - É um campo de estudo da IA voltado para a construção de programas que visam imitar o processo natural da genética. Trabalha com métodos de busca aleatória.
- Raciocínio Baseado em Casos - É o campo de estudo da IA que utiliza uma grande biblioteca de casos para consulta e resolução de problemas. Os problemas atuais são resolvidos através da recuperação e consulta de casos já solucionados e da conseqüente adaptação das soluções encontradas. Por exemplo, o Sistema CASEY,

que faz o diagnóstico em pacientes cardíacos baseado na consulta de arquivos de pacientes com o mesmo diagnóstico.

- **Redes Neurais Artificiais (RNA)** - Possui várias denominações, dentre elas redes neuronais, modelo conectista, neurocomputação, modelo de processamento paralelo distribuído, sistemas neuromórficos e computadores biológicos.
- **Agentes Inteligentes** – um agente é um sistema computacional que, para além das características básicas de autonomia, persistência e sociabilidade, também apresenta características usualmente atribuídas aos humanos, como aprendizagem e capacidades cognitivas.

Existem muitos outros campos de estudo que são englobados no desenvolvimento da IA. Por ser uma ciência relativamente nova possui um potencial muito grande ainda inexplorado, certamente muitas outras áreas de pesquisa e de aplicações deverão surgir nas próximas décadas.

Esses programas já são utilizados com significativa eficiência substituindo ou auxiliando o homem em tarefas de aprendizagem, diagnóstico médico, reconhecimento de padrões, previsões econômicas, tradução, reconhecimento de padrões de imagens e voz, automação de processos industriais, controle de qualidade, etc.

2.3 HEURÍSTICA

As heurísticas ou algoritmos heurísticos foram desenvolvidos com a finalidade de se resolver problemas de elevado nível de complexidade em tempo computacional razoável. Ao se pensar em um problema altamente combinatório, uma opção seria analisar todas as combinações possíveis para conhecer a melhor. Se o problema possui um universo de dados pequeno, realmente esta é a maneira correta de se buscar a melhor solução, mas os problemas reais, normalmente, possuem um número de combinações muito extenso, o que torna inviável a análise de todas as combinações, uma vez que o tempo computacional exigido fica impraticável. As heurísticas procuram encontrar soluções próximas da

otimalidade em um tempo computacional razoável, sem, no entanto, conseguir definir se esta é a solução ótima, nem quão próxima ela está da solução ótima. (CHAVES, 2003)

2.4 ALGORITMOS GENÉTICOS

AGs constituem uma ferramenta poderosa com aplicação em problemas complexos cujos espaços de busca das soluções ótimas podem ser absurdamente grandes para que seja possível determiná-las com precisão através de um método direto. Essas soluções ótimas, em alguns casos, podem nem sequer existir e, muitas vezes, o que procuramos é meramente uma aproximação que nos dê um resultado satisfatório dentro do contexto do problema. É o caso, por exemplo, de jogos, os quais simulam ambientes complexos em que, na maioria das vezes, não há estratégia vencedora (ou não apenas uma única). AGs têm sido aplicados com sucesso em diversos gêneros, desde Otelo até jogos de ação. Outros casos particulares de aplicações dos AGs vêm ganhando interesse nos últimos anos, em especial a *coevolução*, ou seja, populações que evoluem simultaneamente, tanto para *cooperarem* entre si como para *competirem*. Essas possibilidades são muito atraentes para aplicação em jogos, uma vez que os agentes controlados pelo programa estão em constante competição contra o jogador humano com o objetivo de vencê-lo e, ao mesmo tempo, cooperam entre si com a mesma finalidade. (DEMASI, 2002)

2.4.1 Principais Conceitos

Estes são alguns conceitos e definições que serão usados com frequência ao definirmos algoritmos genéticos e sua metodologia:

Cromossomo/ Indivíduos (genótipo) - cadeia de bits que representa uma solução possível para o problema, cada String é um indivíduo.

gene - representação de cada parâmetro de acordo com o alfabeto utilizado (binário, inteiro ou real).

Função de Fitness – parâmetro dos operadores e guia para a obtenção de novas populações.

fenótipo - cromossomo codificado

população - conjunto de pontos (indivíduos) no Espaço de Busca

geração - iteração completa do AG que gera uma nova população

aptidão máxima - melhor indivíduo da população corrente

Deve ser observado que cada cromossomo, chamado de indivíduo no AG, corresponde a um ponto no espaço de soluções do problema de otimização. O processo de solução dotado nos algoritmos genéticos consiste em gerar, através de regras específicas, um grande número de indivíduos, população, de forma a promover uma varredura tão extensa quanto necessária do espaço de soluções.

2.4.2 Conceito de Algoritmo Genético

Algoritmos Genéticos, são métodos de otimização e busca inspirados nos mecanismos de evolução de populações de seres vivos. Foram introduzidos por John Holland e popularizados por seus alunos, David Goldberg. (CARVALHO, 1999)

Os AGs são sistemas adaptativos que imitam alguns mecanismos observados na evolução natural das espécies. Os primeiros passos neste campo foram dados por John Holland no início do anos 70, que acreditou ser possível resolver alguns problemas através de processos de evolução semelhantes. Da Genética sabe-se que os seres vivos têm, no núcleo das suas células, moléculas de DNA onde estão gravados as suas características. Estas moléculas são por sua vez constituídas por cromossomos. Em cada cromossomo encontram-se os genes organizados em cadeias. (RAMOS; SALVADO, 2001)

Durante a reprodução sexual existe troca de material genético entre os dois progenitores, originando novos cromossomos que combinam as características dos cromossomos pais.

Menos frequentemente ocorrem mutações ao nível dos genes. São pequenas alterações acidentais ou induzidas, que poderão ter repercussões nas características do novo ser. O fenómeno da mutação tem uma probabilidade muito reduzida. Quando acontece, o gene de um filho fica diferente do gene correspondente de ambos os pais. Uma reprodução sem mutações implica que cada gene do cromossomo do filho seja igual ao correspondente gene de um dos pais.

Da evolução natural das espécies, sabe-se que os indivíduos mais adaptados ao meio ambiente têm maior probabilidade de sobreviverem e de se reproduzirem. Assim, a sua informação genética será herdada por mais descendentes e terá menor probabilidade de se perder.

Estas características retiradas da própria evolução das espécies serviram de inspiração para a criação dos algoritmos genéticos. Usando seqüências binárias que se designam por cromossomos, e que representam possíveis soluções para um problema, simula-se a evolução natural criando novas seqüências através da reprodução das já existentes. (RAMOS; SALVADO, 2001)

Os algoritmos Genéticos formam a parte da área dos Sistemas Inspirados na Natureza, simulando os processos naturais e aplicando-os à solução de problemas reais. São métodos generalizados de busca e otimização que simulam os processos naturais de evolução, aplicando a idéia darwiniana de seleção. De acordo com a aptidão e a combinação com outros operadores genéticos, são produzidos métodos de grande robustez e aplicabilidade. Estes algoritmos estão baseados nos processos genéticos dos organismos biológicos, codificando uma possível solução a um problema de "cromossomo" composto por cadeia de bits e caracteres. Estes cromossomos representam indivíduos que são levados ao longo de várias gerações, na forma similar aos problemas naturais, evoluindo de acordo com os princípios de seleção natural e sobrevivência dos mais aptos, descritos pela primeira vez por Charles Darwin em seu livro "Origem das Espécies". Emulando estes processos, os Algoritmos Genéticos são capazes de "evoluir" soluções de problemas do mundo real. (MAIA, 1998)

Na natureza os indivíduos competem entre si por recursos como comida, água e refúgio. Adicionalmente, entre os animais de uma mesma espécie, aqueles que não obtêm êxito tendem provavelmente a ter um número reduzido de descendentes, tendo portanto

menor probabilidade de seus genes serem propagados ao longo de sucessivas gerações. A combinação entre os genes dos indivíduos que perduram na espécie, podem produzir um novo indivíduo muito melhor adaptado às características de seu meio ambiente. Os Algoritmos Genéticos utilizam uma analogia direta deste fenômeno de evolução na natureza, onde cada indivíduo representa uma possível solução para um problema dado. A cada indivíduo se atribui uma pontuação de adaptação, dependendo da resposta dada ao problema por este indivíduo. Aos mais adaptados é dada a oportunidade de reproduzir-se mediante cruzamentos com outros indivíduos da população, produzindo descendentes com características de ambas as partes. Se um Algoritmo Genético for desenvolvido corretamente, a população (conjunto de possíveis respostas) convergirá a uma solução ótima para o problema proposto. Os processos que mais contribuem para a evolução são o *crossover* e a adaptação baseada na *seleção/reprodução*. A *mutação* também tem um papel significativo, no entanto, seu grau de importância continua sendo assunto de debate. (MAIA, 1998)

O Algoritmo Genético pode convergir em uma busca de azar, porém sua utilização assegura que nenhum ponto do espaço de busca tem probabilidade zero de ser examinado. As técnicas de busca e otimização tradicionais iniciam-se com um único candidato que, iterativamente, é manipulado utilizando algumas heurísticas (estáticas) diretamente associadas ao problema a ser solucionado. Por outro lado, as técnicas de computação evolucionária operam sobre uma população de candidatos em paralelo. Assim, elas podem fazer a busca em diferentes áreas do espaço de solução, alocando um número de membros apropriado para a busca em várias regiões. Os Algoritmos Genéticos (AGs) diferem dos métodos tradicionais de busca e otimização, principalmente em quatro aspectos: (MAIA, 1998)

1. AGs trabalham com uma codificação do conjunto de parâmetros e não com os próprios parâmetros;
2. AGs trabalham com uma população e não com um único ponto;
3. AGs utilizam informações de custo ou recompensa e não derivadas ou outro conhecimento auxiliar;
4. AGs utilizam regras de transição probabilísticas e não determinísticas.

Algoritmos Genéticos são muito eficientes para busca de soluções ótimas, ou aproximadamente ótimas, em uma grande variedade de problemas, pois não impõem muitas das limitações encontradas nos métodos de busca tradicionais. Os pesquisadores referem-se a "algoritmos genéticos" ou a "um algoritmo genético" e não "ao algoritmo genético", pois AGs são uma classe de procedimentos com muitos passos separados, e cada um destes passos possui muitas variações possíveis. Os AGs não são a única técnica baseada em uma analogia da natureza. Por exemplo, as Redes Neurais estão baseadas no comportamento dos neurônios do cérebro, podendo ser utilizadas em uma grande variedade de problemas de classificação, como reconhecimento de padrões no processo de imagens.

2.4.3 Operadores Genéticos

São os operadores genéticos que transformam a população através de sucessivas gerações, estendendo a busca até chegar a um resultado satisfatório. Um algoritmo genético padrão evolui, em suas sucessivas gerações, mediante o uso de três operadores básicos: (SHAPIRO, 1999)

- *Seleção*
- *Cruzamento Cross-Over*
- *Mutação e Inversão*

A seleção, inspirada no processo de seleção natural dos seres vivos, seleciona os melhores cromossomos da população inicial que é gerada aleatoriamente. O cruzamento é o responsável pela convergência dos AGs. A mutação tem como objetivo introduzir e manter a variedade genética. O operador inversão usa apenas um *cromossomo* para gerar um novo, invertendo a ordem de alguns bits.

2.4.3.1 Cruzamento Cross-Over

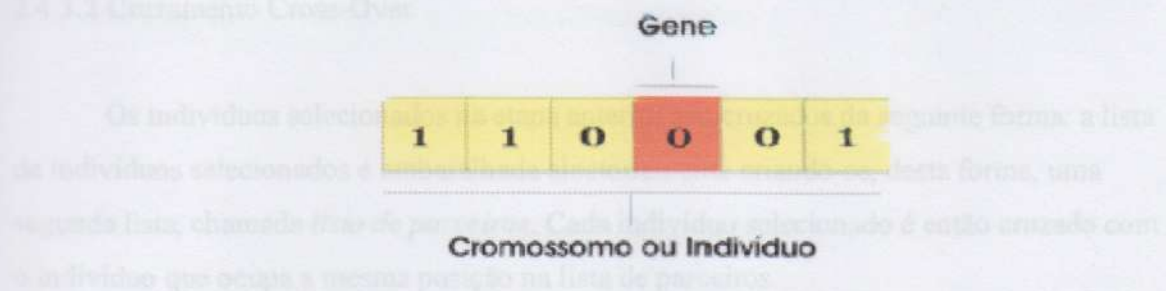


Figura 1 - Exemplo de cromossomo (SOBRINHO, 2004)

2.4.3.1 Seleção

A idéia principal do operador de seleção em um algoritmo genético é oferecer aos melhores indivíduos da população corrente, preferência para o processo de reprodução, permitindo que estes indivíduos passem as suas características às próximas gerações. Isto funciona como na natureza, onde os indivíduos altamente adaptados ao seu ambiente possuem naturalmente mais oportunidades para reproduzir do que aqueles indivíduos considerados mais fracos. Para visualizar este método considere um círculo dividido em n regiões (tamanho da população), onde a área de cada região é proporcional à aptidão do indivíduo (figura 2). Coloca-se sobre este círculo uma "roleta" com n cursores, igualmente espaçados. Após um giro da roleta a posição dos cursores indica os indivíduos selecionados. Este método é denominado *amostragem universal estocástica*. Evidentemente, os indivíduos cujas regiões possuem maior área terão maior probabilidade de serem selecionados várias vezes. Como consequência, a seleção de indivíduos pode conter várias cópias de um mesmo indivíduo enquanto outros podem desaparecer.



Figura 2 - Amostragem estocástica universal

2.4.3.2 Cruzamento Cross-Over

Os indivíduos selecionados na etapa anterior são cruzados da seguinte forma: a lista de indivíduos selecionados é embaralhada aleatoriamente criando-se, desta forma, uma segunda lista, chamada *lista de parceiros*. Cada indivíduo selecionado é então cruzado com o indivíduo que ocupa a mesma posição na lista de parceiros.

O cruzamento é o responsável pela criação de novos cromossomos através da recombinação do material genético de dois cromossomos já existentes na população. Dos vários tipos de cruzamento existentes salientam-se três que se podem considerar os mais genéricos – cruzamento *uniponto*, cruzamento *multiponto* e cruzamento *uniforme*.

Na Figura 3 pode observar-se a representação *uniponto*. Os cromossomos de cada par de indivíduos ao serem cruzados são particionados em um ponto, chamado *ponto de corte*, sorteado aleatoriamente. Um novo cromossomo é gerado permutando-se a metade inicial de um cromossomo com a metade final do outro. Os filhos herdam os bits de um dos pais até ao ponto de cruzamento e a partir deste, os bits de outro pai.

Deve-se notar que se o cromossomo for representado por uma cadeia de bits, o ponto de corte pode incidir em qualquer posição (bit) no interior de um gene, não importando os limites do gene. No caso de genes representados por números reais, a menor unidade do cromossomo que pode ser permutada é o gene.

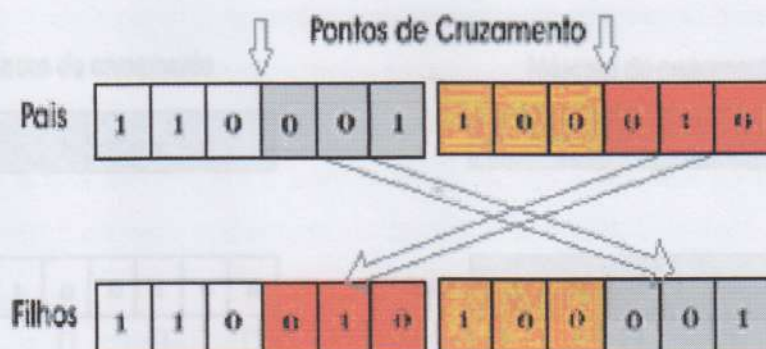


Figura 3 - Exemplo de cruzamento *uniponto*.

Semelhante ao cruzamento anterior existem os cruzamentos *multiponto* com vários pontos de cruzamento conforme se pode observar na Figura 4.

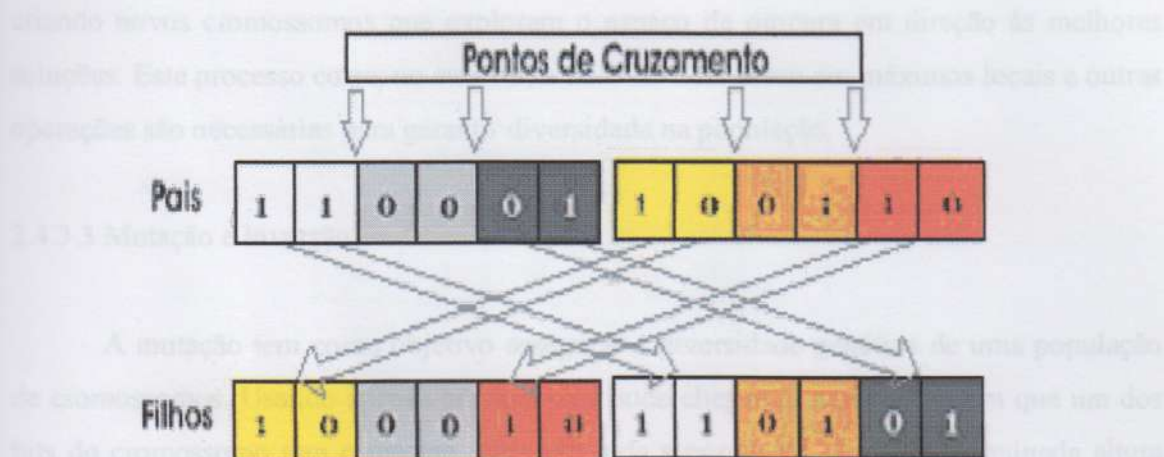


Figura 4 - Exemplo de cruzamento *multiponto*.

No cruzamento *uniforme*, é gerada uma máscara aleatória que indica quais os genes dos dois cromossomos que vão ser trocados (Figura 5).

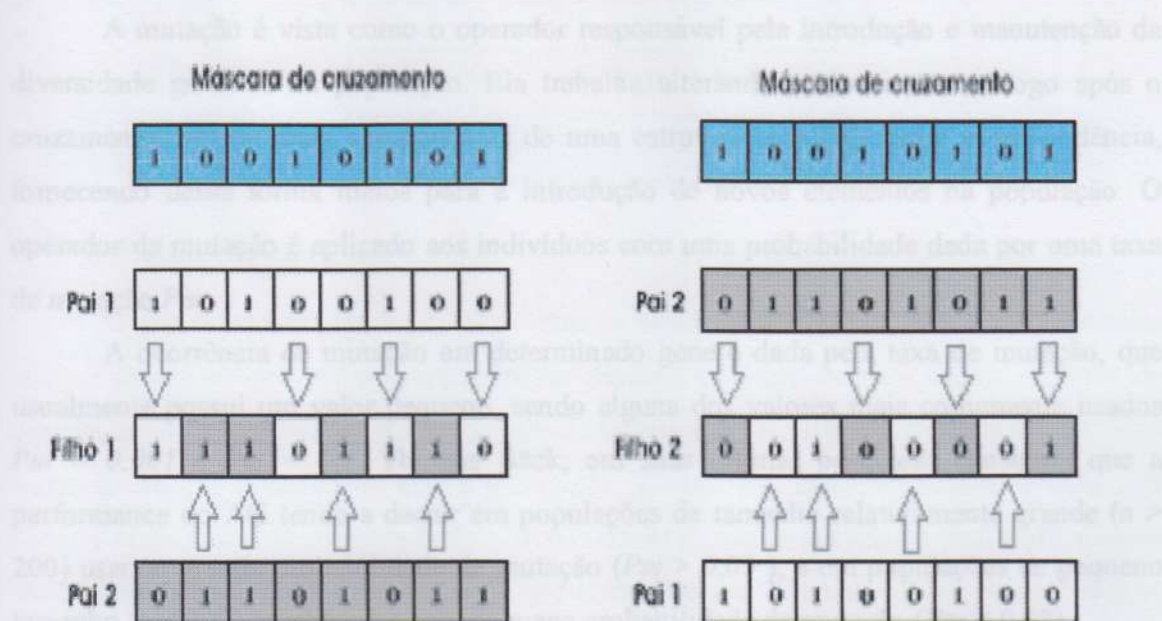


Figura 5 - Exemplo de cruzamento *uniforme*.

O cruzamento é considerado como o verdadeiro responsável pela evolução da qualidade das populações de cromossomos. O fato de recombinar os bits dos melhores cromossomos implica que as melhores seqüências genéticas da população se propaguem

criando novos cromossomos que exploram o espaço de procura em direção às melhores soluções. Este processo corre, no entanto, o risco de ficar preso em máximos locais e outras operações são necessárias para garantir diversidade na população.

2.4.3.3 Mutação e Inversão

A mutação tem como objetivo assegurar a diversidade genética de uma população de cromossomos. Usando apenas cruzamentos pode chegar-se a um ponto em que um dos bits do cromossomo tem o mesmo valor em toda a população. Se em determinada altura todos os cromossomos tiverem o primeiro bit igual a 1 e a melhor solução tiver necessariamente esse bit a 0, o AG, usando cruzamentos, nunca atingirá essa solução. Esse problema pode ser resolvido com mutação que, normalmente com uma probabilidade muito baixa, altera o valor de um bit. Por outro lado um valor muito alto da mutação pode danificar boas seqüências genéticas e tornar o AG numa procura quase aleatória.

A mutação é vista como o operador responsável pela introdução e manutenção da diversidade genética na população. Ela trabalha alterando arbitrariamente, logo após o cruzamento, um ou mais componentes de uma estrutura escolhida entre a descendência, fornecendo dessa forma meios para a introdução de novos elementos na população. O operador de mutação é aplicado aos indivíduos com uma probabilidade dada por uma taxa de mutação P_m .

A ocorrência de mutação em determinado gene é dada pela taxa de mutação, que usualmente possui um valor pequeno, sendo alguns dos valores mais comumente usados $P_m = 0,001$ e $P_m = 0,1$. Thomas Bäck, em suas últimas pesquisas constatou que a performance do AG tende a decair em populações de tamanho relativamente grande ($n > 200$) usando grande probabilidade de mutação ($P_m > 0,05$), e em populações de pequeno tamanho ($n < 200$) combinadas com pequena probabilidade de mutação ($P_m < 0,02$).

2.4.5 O Algoritmo

A seguir temos um Algoritmo genético, com suas principais funcionalidades.

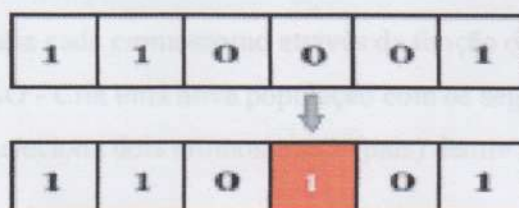


Figura 6 – Exemplo de mutação.

O operador inversão atua apenas num cromossomo invertendo a ordem dos bits entre dois escolhidos aleatoriamente.

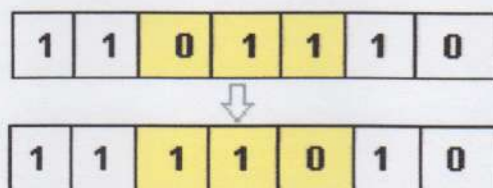


Figura 7 – Exemplo de inversão.

2.4.4 Critério de Parada

Como estamos tratando de problemas de otimização, o ideal seria que o algoritmo terminasse assim que o ponto ótimo fosse descoberto. Mas, existem vários critérios de parada como: 1. quando o AG atingir um dado número de gerações; 2. chegada ao valor ótimo da função objetivo, se este é conhecido; 3. convergência, isto ocorre, quando não ocorrer melhoramento significativo no cromossomo de maior aptidão por um dado número de gerações; 4. o tempo de execução, determina-se um intervalo de repetições; dentre outros.

2.4.5 O Algoritmo

A seguir temos um Algoritmo genético, com suas principais funcionalidades.

1. INICIALIZAÇÃO - Gera aleatoriamente uma população de N cromossomos (soluções convenientes para o problema).
 2. AVALIAÇÃO - Avalia cada cromossomo através da função objetivo (grau de adaptação)
 3. NOVA POPULAÇÃO - Cria uma nova população com os seguintes passos:
 4. SELEÇÃO - seleciona dois cromossomos (pais) dentre a população de acordo com seu grau de adaptação.
 5. CROSSOVER - cruza os pais para formar um novo descendente (filhos).
 6. MUTAÇÃO - altera algum cromossomo da nova geração
 7. ACEITAÇÃO - introduz a nova geração na atual população
 8. SUBSTITUIÇÃO - Utilize a nova população gerada para outro funcionamento do algoritmo
 9. TESTE - se o critério de parada é alcançado, terminar. Caso contrário voltar ao passo 2.
- (ANNES, 2004)

3. METODOLOGIA

Este estudo classifica-se como Pesquisa Aplicada, proporcionando uma compreensão maior do estudo. Segundo Oliveira (1997), a pesquisa aplicada requer teorias como ponto de partida e, objetiva pesquisar, comprovar ou rejeitar hipóteses e fazer sua aplicação às diferentes necessidades humanas.

Os dados deste estudo classificam-se como Pesquisa Qualitativa, pois será analisada a modelagem de algoritmos genéticos, identificando definições de estratégias, efeitos e resultados para a implementação em jogos. Oliveira (1997) afirma que, a abordagem qualitativa nos leva a uma série de leituras sobre o assunto, para estabelecer relações e, a partir daí, darmos nossa conclusão.

Esta pesquisa classifica-se, de acordo com seus objetivos, como uma Pesquisa Descritiva, pois irá descrever o funcionamento dos algoritmos genéticos, em jogos.

Para entender o funcionamento dos AGs em jogos, foi realizado um estudo utilizando dois jogos como exemplo: o *jogo do pingüim e da abelha*, e o *xadrez*, por serem jogos com estratégias e métodos distintos, possibilitando um melhor entendimento no uso das AGs em jogos.



Figura 3 – Interface do jogo do pingüim e da abelha (SILVEIRA, 1998)

O *Jogo do Pingüim e da Abelha* possui a seguinte configuração: uma abelha, um pingüim, cubos de gelo e diversos diamantes. Tanto a abelha quanto o pingüim podem mover-se pela tela (zona de movimentação) de maneira aleatória ou através de um algoritmo genético. Isto é definido pelas opções disponíveis existentes na interface (explicar como a

4. ANÁLISE DE RESULTADOS

Foram escolhidos dois jogos onde os Algoritmos Genéticos são utilizados no desenvolvimento, o do pingüim e da abelha, que é um jogo infantil, e que foi desenvolvido apenas com intuito de observar a funcionalidade dos AGs, é um jogo fácil e com poucos objetos; o outro jogo escolhido, foi o xadrez, por ser mais complexo e trazer outra forma de utilização dos AGs.

4.1 O JOGO DO PINGÜIM E DA ABELHA

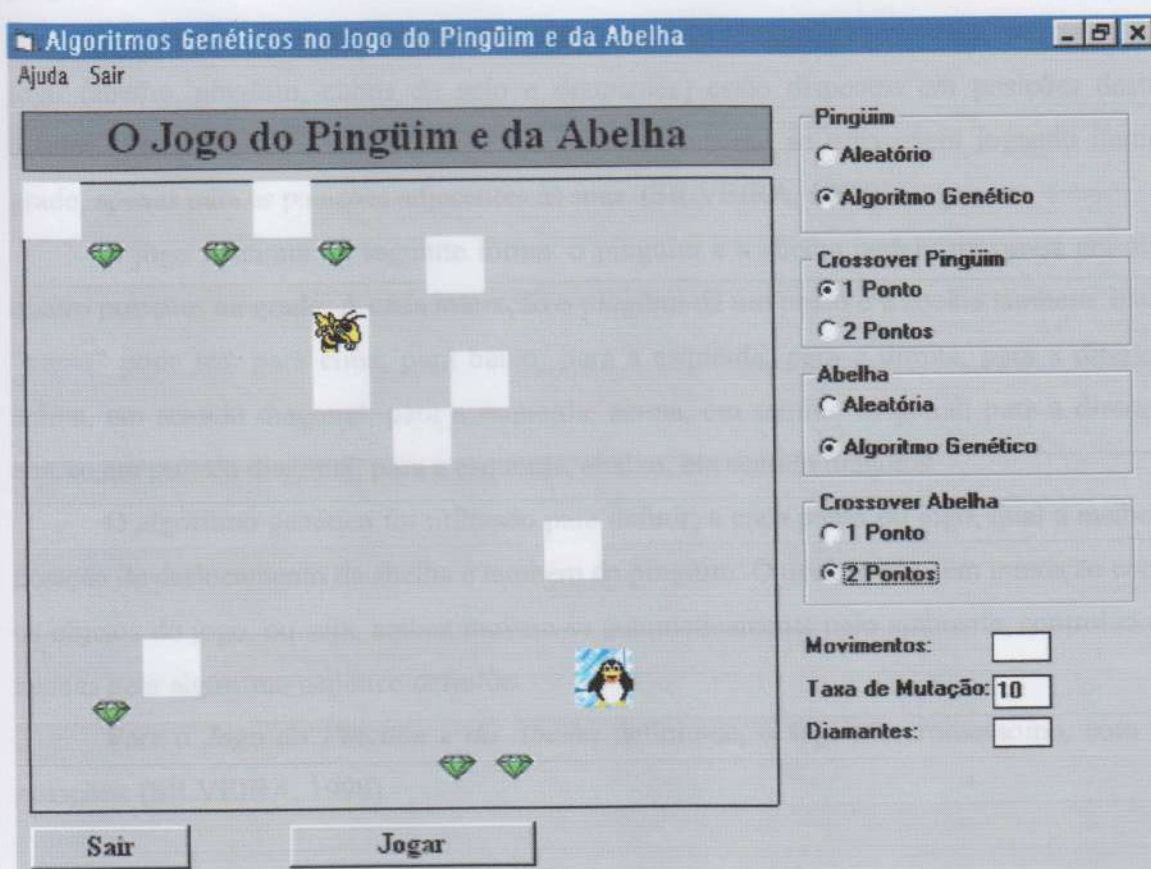


Figura 8 – Interface do jogo do pingüim e da abelha (SILVEIRA, 1998)

O *Jogo do Pingüim e da Abelha* possui a seguinte configuração: uma abelha, um pingüim, cubos de gelo e diversos diamantes. Tanto a abelha quanto o pingüim podem mover-se pela tela (área de movimentação) de maneira aleatória ou através de um algoritmo genético. Isto é definido pelos *option buttons* existentes na interface (conforme mostra a

Figura 8). Além disso, o usuário pode definir o tipo de *crossover* de cada um dos elementos (1 ponto ou 2 pontos) e também a taxa de mutação. Como informações de saída o usuário recebe o número de movimentos efetuados até que alguém vença o jogo (a abelha vence quando atinge o pingüim com um cubo de gelo; o pingüim, por sua vez, vence quando "pega" todos os diamantes) e também o número de diamantes que o pingüim pegou até o momento.

Exemplo O comportamento da abelha consiste em perseguir o pingüim. A abelha precisa "empurrar" um cubo de gelo sobre o pingüim para vencer o jogo. Já o comportamento do pingüim é um pouco mais complexo: deve fugir da abelha e pegar todos os diamantes. O ambiente do jogo é composto por um quadro (mundo imaginário) e os componentes do jogo (abelha, pingüim, cubos de gelo e diamantes) estão dispostos em posições deste quadro. Tanto o pingüim como a abelha movem-se como se estivessem jogando numa grade, apenas para as posições adjacentes às suas. (SILVEIRA, 1998)

O jogo funciona da seguinte forma: o pingüim e a abelha podem mover-se em até quatro posições na grade. A cada interação o pingüim dá um passo e a abelha também. Este "passo" pode ser: para cima; para baixo; para a esquerda; para a direita; para a direita, acima, em sentido diagonal; para a esquerda, acima, em sentido diagonal; para a direita, abaixo, em sentido diagonal; para a esquerda, abaixo, em sentido diagonal.

O algoritmo genético foi utilizado para definir, a cada passo do jogo, qual a melhor posição de deslocamento da abelha e também do pingüim. O usuário não tem interação com os objetos do jogo, ou seja, ambos movem-se automaticamente pelo ambiente, controlados apenas pelo algoritmo genético definido.

Para o *Jogo do Pingüim e da Abelha* definiu-se, o seguinte cromossomo, com 7 posições: (SILVEIRA, 1998)

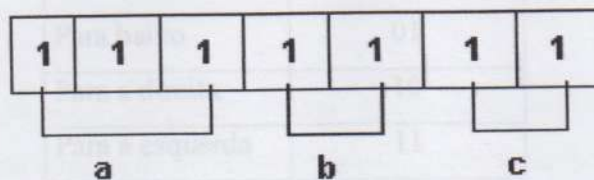


Figura 9 – Cromossomo

Onde:

a = Direção para onde o objeto deve se deslocar

b = Direção para onde o objeto está "olhando" (posicionado)

c = Quantas "casas" o objeto pode se mover na grade (a cada passo).

O cromossomo terá uma codificação binária de acordo com os movimentos possíveis.

Exemplo:

- Direção de deslocamento (a):

Passos	Código
Para cima	001
Para baixo	010
Para a direita	011
Para a esquerda	111
Para a direita abaixo	100
Para a direita acima	101
Para a esquerda acima	110
Para a esquerda abaixo	000

Tabela 1

- Direção de posição, para onde o objeto está olhando (b):

Posição	Código
Para cima	00
Para baixo	01
Para a direita	10
Para a esquerda	11

Tabela 2

- Quantas casas o objeto pode se mover, cada passo (c):

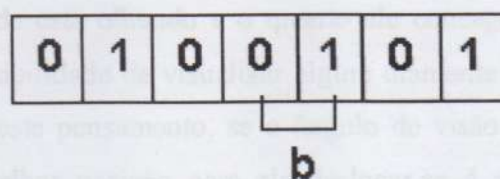
Posição	Código
Um passo	00
Dois passos	01
Três passos	10
Quatro passos	11

Tabela 3

Em uma análise mais detalhada, pode-se notar que a quantidade de bits que cada código ocupa é diretamente proporcional a quantidade de movimentos possíveis em um determinado momento, *a*, *b* ou *c*.

A direção para onde o objeto deve se deslocar (parte *a* do cromossomo) segue a lista de "passos" possíveis para o pingüim e/ou abelha, descritas anteriormente.

A direção para onde o objeto está "olhando" (parte *b* do cromossomo) auxilia na definição do *fitness* do cromossomo. Por exemplo, se a abelha está olhando para baixo (exemplo na figura 10) e o pingüim está posicionado abaixo da mesma, este cromossomo terá um *fitness* mais alto, pois este "passo" da abelha pode fazer com que ela alcance seu objetivo, que é o de "atingir" o pingüim. O cálculo do *fitness* é feito através da direção para onde o objeto está "olhando" e também do número de "casas" que ele pode "andar", definido na parte *c* do cromossomo.

Figura 10 – *fitness* mais alto neste exemplo

O algoritmo genético utilizado possui a seguinte sequência de passos: 1) inicialização aleatória da população de cromossomos; 2) aplicação da função de avaliação; 3) aplicação da mutação e do *crossover* criando novos cromossomos; 4) aplicação da função de avaliação e inserção dos novos cromossomos na população; 5) critério de parada.

1) Inicialização

Na inicialização, o algoritmo gera cadeias de cromossomos aleatórias para cada personagem do jogo, na quantidade pré-determinada pelo algoritmo, como na figura abaixo.

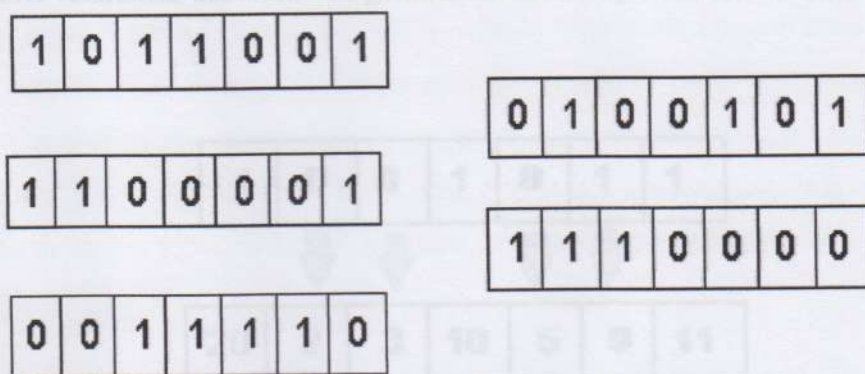


Figura 11 – população inicial

2) Função Avaliação

Tomada como base a população gerada a função de avaliação será definida, a partir da posição do objeto. Por exemplo: para o pingüim, a melhor posição, dentre as selecionadas (dentre os cromossomos selecionados) pelo algoritmo genético, é a que possui um diamante ou está mais distante da abelha. Toma-se necessário, aqui, definir o ângulo de visão do pingüim: para onde está olhando e o quanto ele consegue "enxergar". A partir disso, tem-se ou não a possibilidade de visualizar algum diamante ou a abelha e avaliar a melhor posição. Seguindo este pensamento, se o ângulo de visão da abelha não permite visualizar o pingüim, a melhor posição para ele deslocar-se é a mais próxima de um diamante, pois o objetivo do pingüim, no jogo descrito, é "pegar" todos os diamantes existentes.

A cada passo do jogo (movimento do pingüim e da abelha), tem-se uma nova população de cromossomos, que serão avaliados (retornando o *fitness* de cada cromossomo).

3) Mutação e Crossover

A taxa de mutação é definida pelo usuário, através da interface do jogo (que pode ser vista na interface da figura 8), ou seja, como no exemplo a taxa digitada pelo usuário tem valor 10, e foram Gerados números aleatórios para cada *bit* do cromossomo (figura 12), todos os *bits* referentes aos números gerados aleatórios que forem menores que 10 serão mutados.

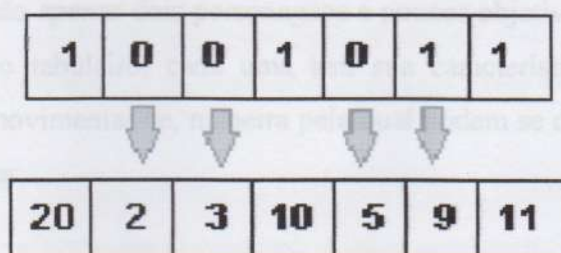


Figura 12 – números criados aleatórios

Para a realização do *crossover* será utilizado o método de cruzamento entre os pais. Este cruzamento poderá ser de um ou dois pontos. O usuário, antes de dar início ao jogo, define se o pinguim e/ou a abelha vão utilizar o *crossover* de 1 ponto (Figura 3) ou o de 2 pontos (multiponto, figura 4), através de *option buttons* dispostos na interface do jogo.

Para a seleção dos pais que irão sofrer o *crossover* utilizou-se a técnica da roleta (figura 2). Inicialmente acumula-se o *fitness* de todos os cromossomos. A partir do *fitness acumulado*, geram-se números aleatórios no intervalo de zero até este *fitness acumulado*, para definir qual cromossomo está mais próximo, ou dentro do intervalo deste número gerado. A partir disso, organiza-se a população em ordem decrescente do *fitness* e cruzam-se os pais pela disposição definida após a organização.

5) Critério de Parada

Podem existir vários critérios de parada do Algoritmo Genético nesse jogo, o mais provável que tenha sido utilizado é atingir um número N de gerações. Exemplo: na implementação foi determinado que o laço de gerações seja finalizado após 8 resultados, então serão geradas 8 soluções independente desde que a primeira já ter sido a melhor. Só

após finalizar a quantidade de vezes pré-determinada, é que será escolhido o mais apropriado dentre os oito encontrados, sendo finalmente realizada a jogada.

4.2 O JOGO DE XADREZ

No jogo de xadrez (RAMOS, 2001), o desenvolvimento é muito mais complexo do que o do pingüim e da abelha, a começar pela quantidade de peças existentes. No jogo do pingüim e da abelha, são apenas dois personagens e poucos objetivos; no xadrez, além das trinta e duas peças no tabuleiro, cada uma tem suas características individuais, como: quantidade de casa a movimentar-se, maneira pela qual podem se deslocar e qual a melhor forma de se movimentar.

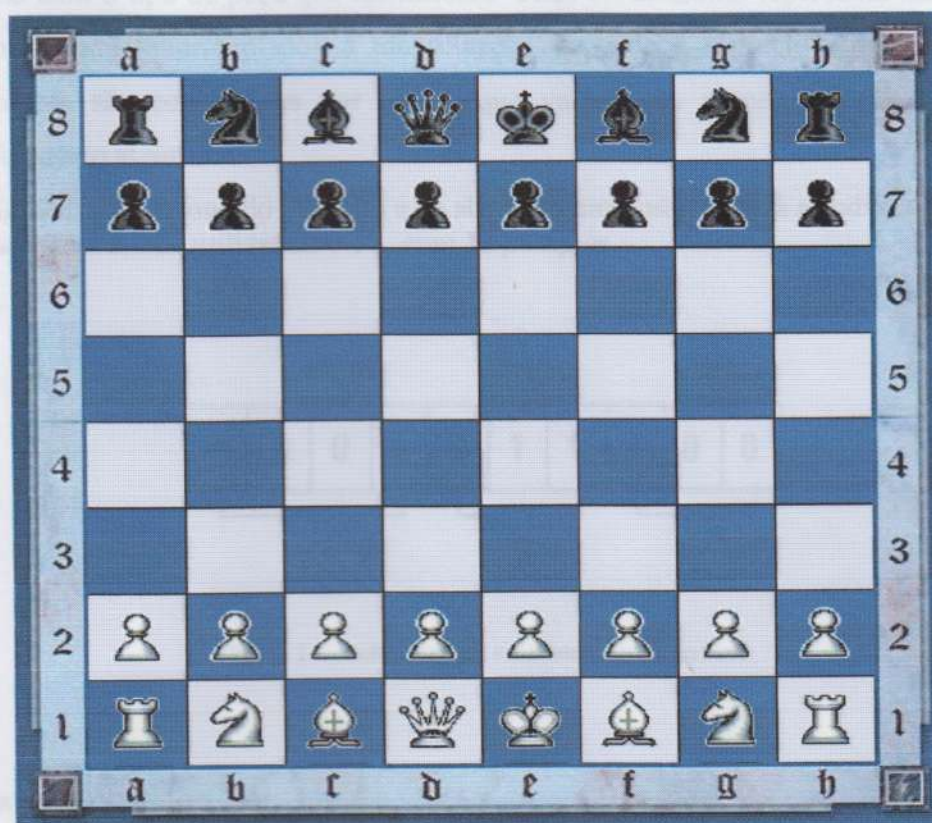


Figura 13 – Interface do jogo xadrez

Cada jogo é constituído por um tabuleiro e dois jogadores, humanos e/ou máquinas, possuindo cada jogador 16 peças. O tabuleiro do jogo de xadrez, encarrega-se de

memorizar as peças que estão em jogo e suas respectivas posições no mesmo. A cada jogador são atribuídas dezesseis peças, de seis tipos diferentes; especificamente um Rei, uma Rainha, dois Bispos, dois Cavalos, duas Torres e oito Peões. (RAMOS, 2001)

Ao tomar uma peça, a peça do jogador ficará no mesmo lugar que se encontrava a do adversário. O objetivo do jogo é deixar o rei sem espaço para se movimentar. Quando o rei estiver ameaçado, está em Xequê; quando não tiver saída, é Xequê Mate.

Os AGs são utilizados nos jogos de xadrez para procurar a estratégia adequada à cada jogada, diante de um conjunto de soluções muito amplo, o que levaria o computador a gastar muito tempo procurando a melhor solução se isto fosse feito analisando jogada a jogada.

Os cromossomos têm representação binária e o seu tamanho e codificação variam de acordo com o tipo de peça que identificam. Cada cromossomo representa um possível lance, onde estão representados a identificação da peça e respectivo movimento. Os cromossomos são constituídos por genes, que variam em número de acordo com a profundidade do algoritmo.

Igualmente ao jogo do pingüim e da abelha, o cromossomo terá a codificação cromossômica binária, utilizando 0 e 1, com 10 posições.

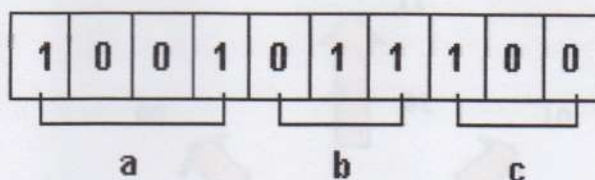


Figura 14 – estrutura do cromossomo no xadrez

Onde:

a = peça do jogo

b = direção que tal peça tomará

c = deslocamento (quantidade de casas a andar)

- Peças do jogo (a):

Código	Peça
0000	Peão 1
0001	Peão 2
0011	Peão 3
0010	Peão 4
0110	Peão 5
0111	Peão 6
0101	Peão 7
0100	Peão 8

Código	Peça
1000	Torre 1
1001	Torre 2
1010	Cavalo 1
1011	Cavalo 2
1100	Bispo 1
1101	Bispo 2
1110	Rainha
1111	Rei

Tabela 4

A codificação do movimento varia conforme o tipo de peça a que se refere o gene.

Começando pelo Peão (Figura 15), são necessários apenas dois bits para codificar os seus movimentos pois esta peça só tem quatro possíveis – avançar uma ou duas casas (01 ou 11) e tomar à esquerda ou à direita (00 ou 10). Os genes que representem movimentos de Peões ficam assim reduzidos a seis bits, quatro para identificação e dois para o movimento.

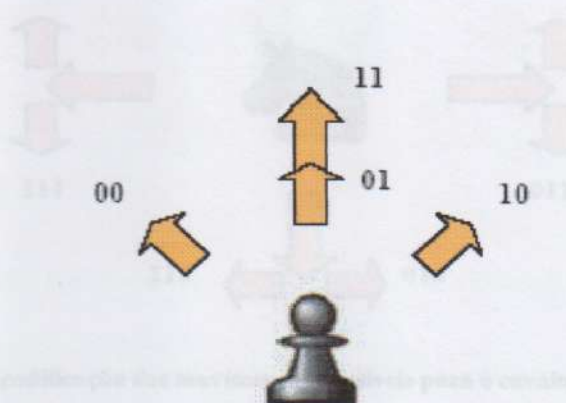


Figura 15 – codificação dos movimentos possíveis do peão (RAMOS, 2001)

Em relação à Torre (Figura 16), o movimento é baseado na direção e no deslocamento. Dois bits são suficientes para codificar as quatro direções possíveis, enquanto que para o deslocamento são necessários três bits. No total, os genes representantes das Torres são constituídos por nove bits.

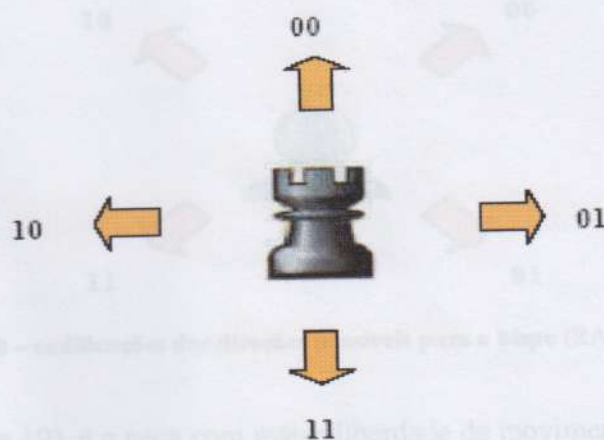


Figura 16 – Codificações das direções possíveis da torre (RAMOS, 2001)

O Cavalo (Figura 17), necessita de apenas 3 bits para codificar os seus oito movimentos possíveis, resultando assim genes de sete bits.

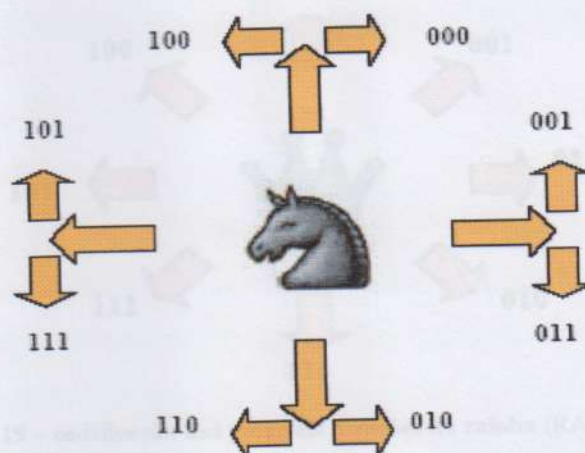


Figura 17 – codificação dos movimentos possíveis para o cavalo (RAMOS, 2001)

O Bispo (Figura 18), é em tudo semelhante à Torre, variando apenas a direção dos seus movimentos. São necessários genes de nove bits para codificar um movimento de um Bispo.

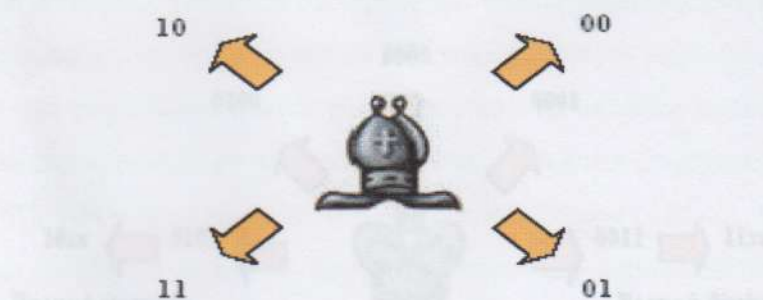


Figura 18 – codificações das direções possíveis para o bispo (RAMOS, 2001)

A Rainha (Figura 19), é a peça com maior liberdade de movimentos, sendo por isso a que necessita de mais bits de codificação. Como tem mais um grau de liberdade na direção que o Bispo e a Torre, necessita de mais um bit para a codificação do movimento, utilizando assim dez bits para codificar os seus genes.

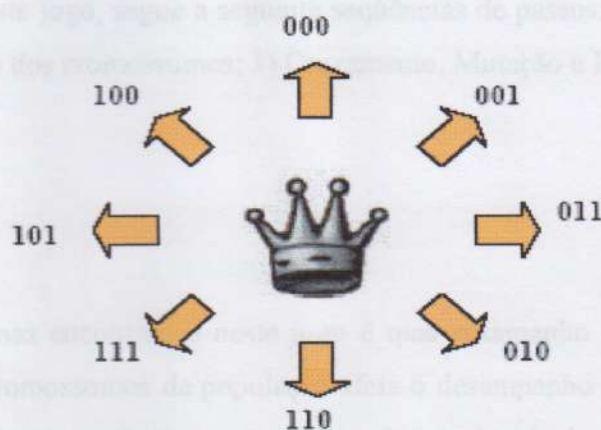


Figura 19 – codificação das possíveis direções da rainha (RAMOS, 2001)

O Rei (Figura 20), tem dez movimentos diferentes. Além dos oito correspondentes a uma casa em todas as direções tem mais dois respeitantes ao roque à esquerda e ao roque à direita (dependendo da cor do Rei podem ser considerados roque pequeno ou roque grande) São necessários quatro bits para codificar o seu movimento, que somados com os quatro da identificação, resultam oito bits para a codificação dos genes.

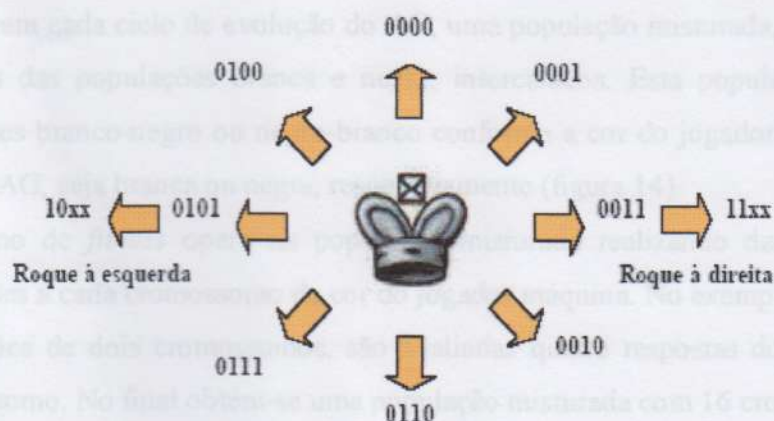


Figura 20 – codificação dos movimentos possíveis para o rei (RAMOS, 2001)

As populações iniciais serão divididas em níveis de profundidade, de acordo com o tamanho do cromossomo.

O AG usado, neste jogo, segue a seguinte seqüências de passos: 1) inicialização da população; 2) Avaliação dos cromossomos; 3) Cruzamento, Mutação e Inversão; 4) Critério de parada.

1) Inicialização

Um dos problemas encontrados neste jogo é qual o tamanho da população a ser gerado. O número de cromossomos da população afeta o desempenho do AG. Populações pequenas podem levar à convergência para máximos locais devido à reduzida diversidade genética. Por outro lado se o tamanho da população for muito grande pode levar o AG a perder tempo desnecessário a explorar o espaço de procura.

Este AG tem na sua base duas populações, uma de lances brancos e outra para os lances negros. Os operadores genéticos, cruzamento e mutação, vão operar nestas duas populações separadamente, originando novas gerações, os chamados filhos da velha geração. A avaliação das gerações é feita tendo em conta não só a qualidade dos lances individuais da respectiva cor, mas também em consideração as possíveis respostas do adversário. Não pode ser considerado um bom lance tomar um Cavalo adversário, colocando em risco uma peça de maior valor como por exemplo uma Dama. A resposta do

adversário é sempre levada em conta. Para se conseguir avaliar estas seqüências de jogada-resposta é criada, em cada ciclo de evolução do AG, uma população misturada, constituída por cromossomos das populações branca e negra, intercalados. Esta população é uma seqüência de lances branco-negro ou negro-branco conforme a cor do jogador máquina, a que pertence este AG, seja branca ou negra, respectivamente (figura 14).

O algoritmo de *fitness* opera na população misturada realizando desta maneira múltiplas avaliações a cada cromossomo da cor do jogador máquina. No exemplo da Figura 14, com populações de dois cromossomos, são avaliadas quatro respostas do adversário para cada cromossomo. No final obtém-se uma população misturada com 16 cromossomos.

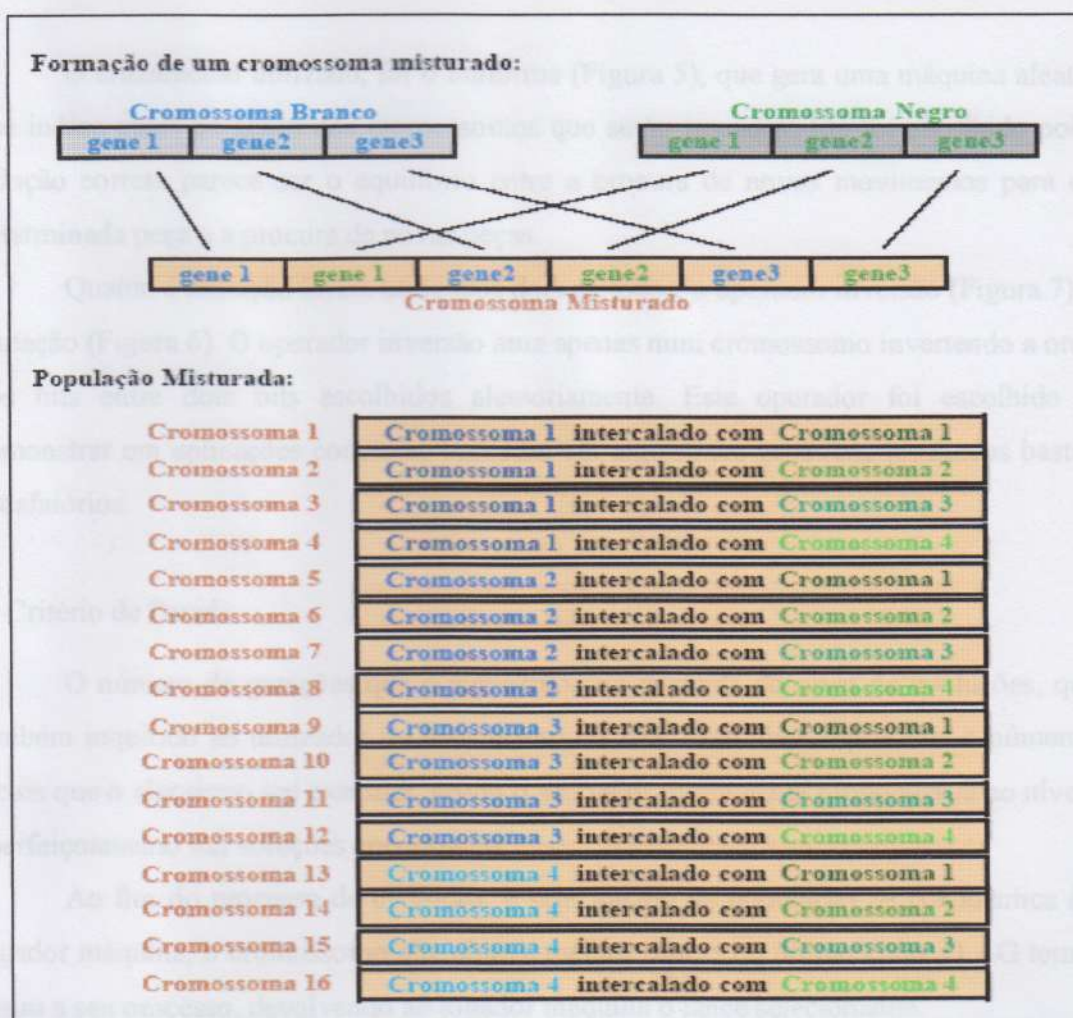


Figura 21 – formação de uma população misturada (RAMOS, 2001)

2) Avaliação dos Cromossomos

Sendo obrigatório a avaliação dos lances com uma resposta do adversário, a utilização de populações separadas para cada cor foi a forma encontrada para evitar uma melhoria dos lances do jogador máquina e degradação das possíveis respostas do adversário. O objetivo centra-se em garantir que o adversário responda sempre da melhor forma.

Serão avaliados os cromossomos respectivos a cor do jogador máquina.

3) Cruzamento, Mutação e Inversão

O cruzamento utilizado, foi o uniforme (Figura 5), que gera uma máquina aleatória que indica quais os genes dos cromossomos que serão trocados. Ele foi escolhido pois, a solução correta parece ser o equilíbrio entre a procura de novos movimentos para uma determinada peça e a procura de novas peças.

Quanto à mutação foram utilizados dois métodos, o operador inversão (Figura 7) e a mutação (Figura 6). O operador inversão atua apenas num cromossomo invertendo a ordem dos bits entre dois bits escolhidos aleatoriamente. Este operador foi escolhido por demonstrar em aplicações concretas baseadas em algoritmos genéticos resultados bastante satisfatórios.

4) Critério de Parada

O número de gerações que o algoritmo cria depende do nível de evoluções, que é também inquirido ao utilizador no lançamento do AG. Esse nível representa o número de ciclos que o algoritmo vai executar, sendo o seu valor diretamente proporcional ao nível de aperfeiçoamento das soluções encontradas.

Ao fim do processo de evolução, é selecionado da população de cor idêntica à do jogador máquina, o cromossomo que obteve melhor *fitness* no último ciclo. O AG termina assim a seu processo, devolvendo ao jogador máquina o lance selecionando.

No turno seguinte, quando o jogador máquina solicitar novamente ao seu AG um lance, o algoritmo começa o seu processo executando um *backup* às populações branca e

negra que concluíram o processo no turno anterior. Deste modo não se perdem possíveis seqüências de lances já estudados pelo AG, com boas hipóteses de obterem resultados satisfatórios, melhorando assim o desempenho do algoritmo.

5. CONCLUSÃO

A idéia inicial deste trabalho, seria a implementação do jogo de xadrez usando Algoritmo Genético, mas devido o curto prazo de tempo, a idéia inicial foi substituída.

Foi feito apenas um estudo teórico do algoritmo usado em dois jogos distintos, que possibilitou o entendimento de como se comportam os operadores genéticos para seleção de estratégias num campo de possibilidades amplo.

Com este trabalho foi possível aprender o funcionamento dos Algoritmos genéticos, e visualizar seu comportamento e aplicação em jogos.

No jogo do pingüim e da abelha, o algoritmo por ter um campo de soluções menor que o xadrez, saiu-se bem, o tempo de execução de cada objeto foi rápido, e com resultados satisfatórios, ambos objetos em momentos aleatórios terminavam vitoriosos, fazendo-se viável para a implementação do mesmo.

Já no jogo de xadrez, os resultados foram diferentes, segundo a observação de Ramos e Salvado (2001), desenvolvedores do software estudado o jogo consegue mostrar durante vários momentos um nível bastante aceitável. No início consegue mesmo surpreender devido à coerência e exactidão dos lances efectuados. Verifica-se no entanto que o nível vai diminuindo gradualmente à medida que se desenrola a partida. De um modo empírico e sem recorrer a nenhum estudo podemos dizer que no início o jogo apresenta um nível de um jogador experiente e à medida que o jogo vai decorrendo esse nível passa a ser de um jogador novato. Deixando assim concluir, que a utilização de algoritmo genético para solucionar qual melhor estratégia no jogo de xadrez, não seja o método mais apropriado.

A principal contribuição deste trabalho é a apresentação de Algoritmos Genéticos em detalhes, descrevendo os principais conceitos, operadores e modo de funcionamento. A aplicação de AGs em jogos deixa claro a forma como eles funcionam, saindo de um contexto puramente teórico para um contexto mais prático, onde observamos com mais clareza o comportamento dos AGs.

Como já foi dito, AGs não podem ser definidos como a melhor solução tratando-se de jogos. Como toda técnica heurística, o problema é quem define a sua adequabilidade ou não a situação.

6. RE Como sugestão de trabalhos futuros, propomos a implementação de Algoritmos Genéticos em alguns jogos de estratégia, para analisar o seu comportamento em tais situações.

Disponível em: <<http://pucrs.campus2.br/~ciencia/ig.html>> Acesso em: 24 Nov. 2004.

BITTENCOURT, Guilherme. *Inteligência Artificial: Fundamentos e Teorias*, Campinas, 1999.

CARVALHO, André C.P.L.F.; GALVÃO, Carlos D.; LACERDA, Edson G. M.; DINIZ, Laudice S.; VALENÇA, Márcio J. S.; LIL-AMOR, Tereza R.; VIEIRA, Vitor P.P.B. *Sistemas Inteligentes: Aplicações e Recursos Híbridos e Ciências Ambientais*, Porto Alegre – Ed. Universidade, 1999.

CHAVES, Arthur A. *Modelagem Formal e Matemática para a Resolução de Problemas do Caminho Viável com Coleta de Preços*. Dissertação (Trabalho Final de Curso) – Universidade Federal do Rio de Janeiro. Profa. Disponível em: <<http://www.dcc.ufpe.br/~profis/pos/Dissertacoes/PCVCP-Eixo-VNS.pdf>> Acesso em: 23 Nov. 2004.

DEMASI, Pedro; CHUZ, Adriano J. O. *Algoritmos Evolucionários Cooperativos em Jogos*. Artigo – Universidade Federal do Rio de Janeiro. Disponível em: http://www.lidiane.org.br/downloads/wagner_evo_2002.pdf Acesso em: 05 Nov. 2004.

LEMA, Cynthia M. *Introdução à Inteligência Artificial*. 2004. Artigo. Disponível em: [cynthia.m.lidiane.org.br/downloads/wagner_evo_2002.pdf](http://www.lidiane.org.br/downloads/wagner_evo_2002.pdf) Acesso em: 15 Out. 2004.

OLIVEIRA, S. L. de. *Tratado de metodologia científica: projetos de pesquisas, TGI, TCC, monografias, dissertações e teses*. São Paulo: Pearson, 1997.

PIAGET, Jean. *Psicologia e Pedagogia*. Trad. Por Doreu Acrioly Linslow e Rosa Maria Ribeiro da Silva. Rio de Janeiro: Tercer Universitária, 1976.

MAIA, Alexandre C. P.; MICHELAN, Roberto. *Algoritmos Genéticos*. 1998. Universidade Estadual de Maringá, Departamento de Informática. Disponível em: <<http://www.dcc.ufpe.br/~profis/pos/Dissertacoes/PCVCP-Eixo-VNS.pdf>> Acesso em: 05 Nov. 2004.

RAMOS, N. SALVADO, S. *Jogo De Xadrez Por Algoritmos Genético-Evolucionários*. 2001 Dissertação (Trabalho Final de Curso) – Instituto Superior Técnico, Lisboa – Portugal. Disponível em: <<http://lancet.isr.isr.utl.pt/publication/ifo/ChessGA/chess.pdf>> Acesso em: 14 Mar. 2004.

SABBATINI, R.M.L. *Um Jogo muito Humano*. 1992. (Informática) – Campinas. Disponível em: <<http://www.cayub.org.br/correio/con91.htm>>. Acesso em: 22 Mar. 2004.

SANTOS, G.L. *Inteligência Artificial em jogos 3D*. 2002. Dissertação (Monografia) – Instituto de Computação, Rio de Janeiro. Disponível em: <

6. REFERÊNCIAS BIBLIOGRÁFICAS

- ANNES, Ricardo. **Inteligência Artificial**. 2004 .Artigo (Aula) – Pontificia Universidade Católica – Uruguaiana. Disponível em: < http://puers.campus2.br/~annes/ia_ag.html > Acesso em: 24 Nov.2004.
- BITTENCOURT, Guilherme. **Inteligência Artificial: Ferramentas e Teorias**, Campinas, 1996.
- CARVALHO, André C.P.L.F.; GALVÃO, Carlos O.; LACERDA, Estéfane G. M.; DINIZ, Laudízio S.; VALENÇA, Mêuser J. S.; LUDERMIR, Tereza B.; VIEIRA, Vicente P.P.B. **Sistemas Inteligentes: Aplicações a Recursos Hídricos e Ciências Ambientais**, Porto Alegre – Ed. Universidade, 1999.
- CHAVES, Antônio A. **Modelagens Exata e Heurística para a Resolução do Problema do Caixeiro Viajante com Coleta de Prêmios**. Dissertação (Trabalho Final de Curso) – Universidade Federal de Ouro Preto. Disponível em: <<http://www.decom.ufop.br/prof/marcone/Orientacoes/PCVCP-Exato-VNS.pdf>> Acesso em: 23 Nov. 2004.
- DEMASI, Pedro; CRUZ, Adriano J. O. **Algoritmos Coevolucionários Cooperativos em jogos**. Artigo – Universidade Federal do Rio de Janeiro. Disponível em: http://www.labic.nce.ufrj.br/downloads/wjogos_cca_2002.pdf> Acesso em: 05 Nov. 2004.
- LIMA, Cynthia M. **Introdução á Inteligência Artificial**. 2004. Artigo. Disponível em: cynthia_m_lima.sites.uol.com.br/ia.html> Acesso em: 15 Out. 2004.
- OLIVEIRA, S. L. de. **Tratado de metodologia científica: projetos de pesquisas, TGI, TCC, monografias, dissertações e teses**. São Paulo: Pioneira, 1997.
- PIAGET, Jean. **Psicologia e Pedagogia**. Trad. Por Dirceu Accioly Lindoso e Rosa Maria Ribeiro da Silva. Rio de Janeiro: Forense Universitária, 1976.
- MAIA, Alexandra C. P.; MICHELAN, Roberto. **Algoritmos Genéticos**. 1998. Universidade Estadual de Maringá. Departamento de Informática. Disponível em: < <http://www.din.uem.br/ia/intelige/geneticos/> >. Acesso em: 05 Nov. 2004.
- RAMOS, N.;SALVADO, S. **Jogo De Xadrez Por Algoritmos Genético-Evolutivos**. 2001.Dissertação (Trabalho Final de Curso) – Instituto Superior Técnico. Lisboa - Portugal Disponível em: <<http://laseeb.isr.ist.utl.pt/publications/tfc/ChessGA/chess.pdf>>. Acesso em: 14 Mar. 2004.
- SABBATINI, R.M.E. **Um jogo muito humano**. 1992. (Informática) – Campinas. Disponível em: < <http://www.epub.org.br/correio/corr91.htm> >. Acesso em: 22 Mar. 2004.
- SANTOS, G.L. **Inteligência Artificial em jogos 3D**. 2002. Dissertação (Monografia) – Instituto de Computação, Rio de Janeiro. Disponível em: <

[http://www.paralelo.com.br/documentos/Real-Time%20BSP-Based%20Path%20Planning%20with%20A-Star%20\(Full-Portuguese\).pdf](http://www.paralelo.com.br/documentos/Real-Time%20BSP-Based%20Path%20Planning%20with%20A-Star%20(Full-Portuguese).pdf)>. Acesso em: 06 Abr. 2004.

SHAPIRO, J. **Genetic Algorithms in Machine Learning**, taught at the Advanced Summer School on Machine Learning and Applications, 1999.

SILVEIRA, Sidnei R. **Jogos Educativos Computadorizados utilizando a Abordagem de Algoritmos Genéticos**, 1998. Artigo (Pós-Graduação), Universidade Federal do Rio Grande do Sul. Porto Alegre. Disponível em: <
<http://www.c5.cl/ieinvestiga/actas/ribie98/151.html> > Acessado em: 22 Out. 2004.

SOBRINHO, Antônio C. C. S.; GIRARDI, Maria R. **Uma Análise das Aplicações dos Algoritmos Genéticos em Sistemas de Acesso à Informação Personalizada**. 2004. Artigo. Universidade Federal do Maranhão. São Luís. Disponível em: <
<http://www.sbc.org.br/reic/edicoes/2003e4/tutoriais/AlgoritmosGeneticosEmSistemasDeAcessoAInformacaoPersonalizada.pdf> > Acesso em: 01 nov. 2004.

TATAI, Victor K. **Técnicas de Sistemas Inteligentes Aplicadas ao Desenvolvimento de Jogos de Computador**, 1992. Dissertação (Tese de Mestrado), Universidade Estadual de Campinas. Disponível em: <